

**Projeto Final**

**10. Relógio Despertador Programável**

Departamento de Eletrónica, Telecomunicações e Informática

Mestrado Integrado em Engenharia Eletrónica e Telecomunicações

Ano Letivo 2018/2019

Laboratórios de Sistemas Digitais

P14 – Prof. José Luís Azevedo

Nome: Emanuel Veiga Pereira

Nmec: 93235;

03/06/2018

**Introdução**

O trabalho consiste na modelação de hardware de um relógio despertador programável, tendo como exemplo os relógios do nosso quotidiano, com as funções: relógio normal, acerto ou configuração de horas/dias da semana e despertador programável. (R.D.P = relógio despertador programável)

**Arquitetura do sistema**

Neste projeto é necessário usar módulos como por exemplo:

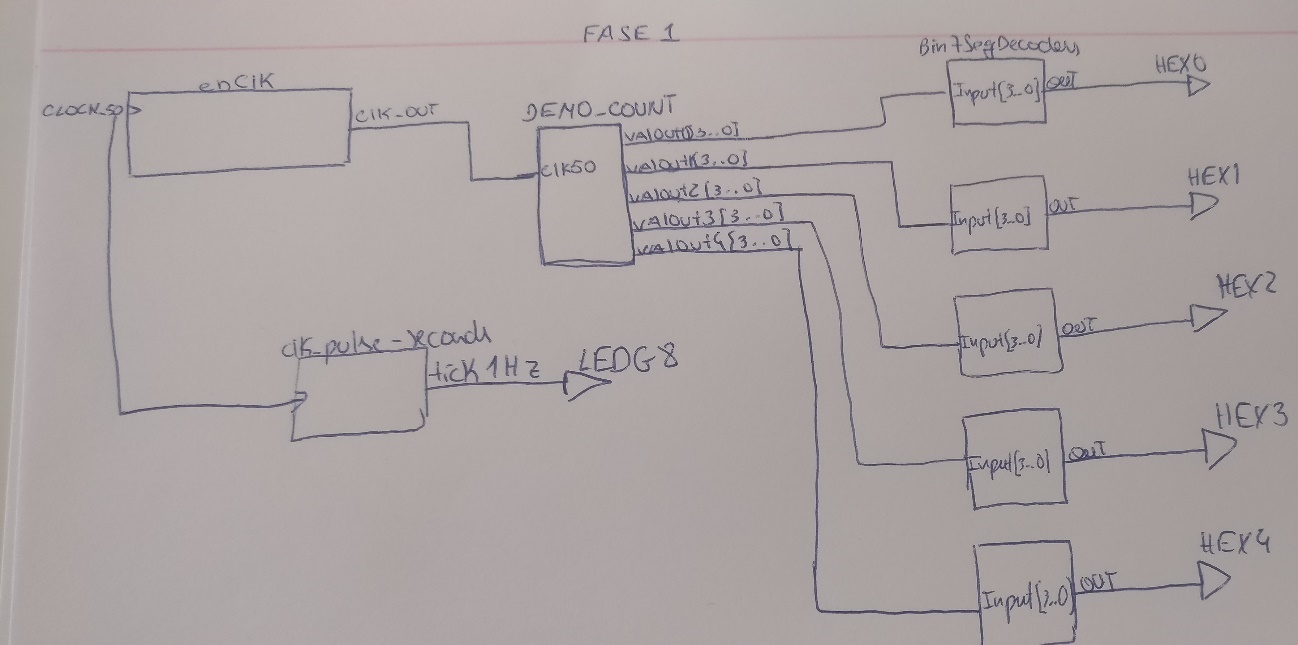
 **Relógio normal**, quando o modo de configuração apresenta as entradas “00” inserida pelo utilizador: Neste caso 7 contadores que servem como relógio (um contador para os minutos unidades, minutos dezenas, horas unidades, horas dezenas e dia da semana e outros dois contadores para os segundos mas em modo open, ou seja invisíveis) e um sinal com frequência de 1Hz, vindo de um pulse generetor que serve para os segundos. A cada pulso de relógio deverá ser incrementado 1 minuto;

Figura 1: Relógio normal

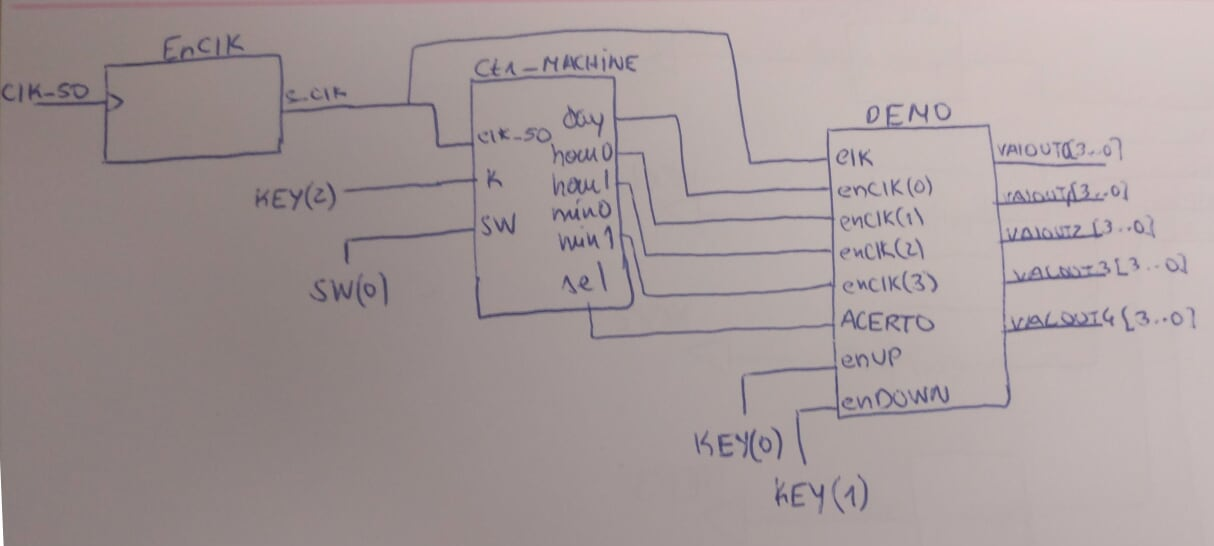
**Relógio ajustável** terá as mesmas funcionalidades que o relógio normal sendo que o utilizador poderá selecionar o dígito que pretende alterar, de uma forma sequencialmente, através de uma máquina de estados. Sendo que em cada dígito poderá ter a opção de aumentar o dígito ou diminuir, através de um novo contador acrescentando as condições de enUp e enDown e modo Acerto. O DeBouncer foi usado na KEY(2);

Figura 2: Relógio normal ajustável

**Relógio Despertador Programável:** o sistema vai ter uma memória RAM que armazenará os alarmes inseridos pelo utilizador, neste momento estou a testar ainda com register para armazenar o valor inserido pelo utilizador, mais tarde passará para uma RAM;

**Implementação**

**Unidade de Controlo:** A unidade de controlo é essencialmente uma máquina de estados que controla o que aparece nos displays e modos de configuração.

Fase2(Unidade de Controlo): Esta fase é composta por 6 estados, um estado para quando está a contas (running) e cinco estados para cada digito, para fins de acertar o relógio;

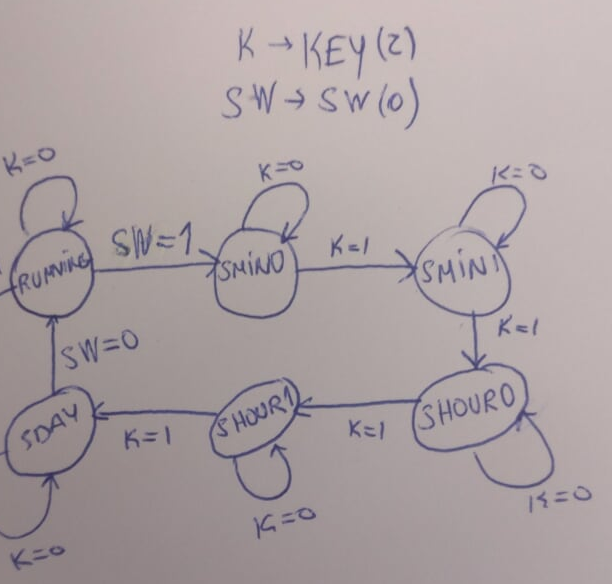


Figura 3: Máquina de Estados da Fase2

Fase3(Unidade de Controlo): Esta fase é composta por mais um estado (Alarm) que permite ao utilizador guardar um alarme;

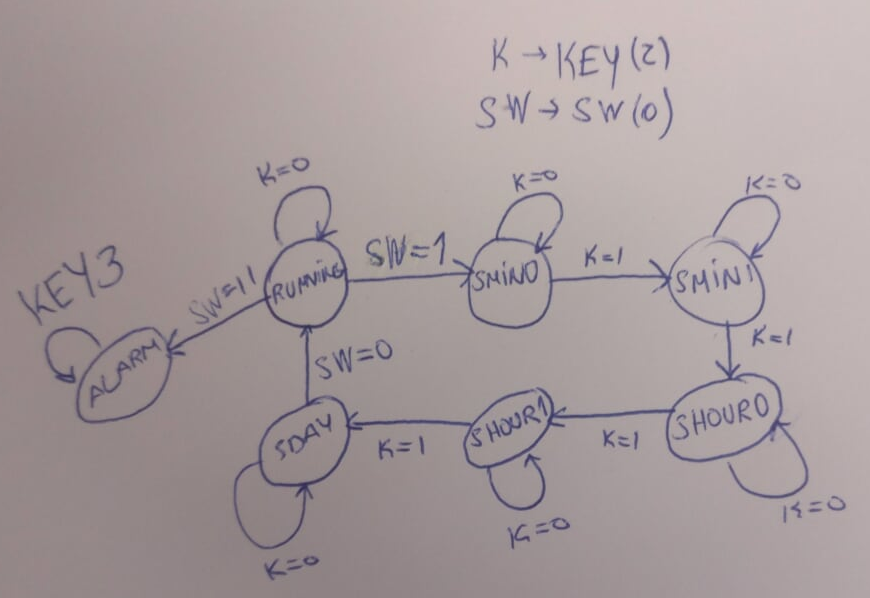


Figura 4: Máquina de Estados da Fase3

**Validação**

As validações deste projeto foram maioritariamente feitas através de testes no kit da FPGA, sendo que utilizei várias vezes o VWF para acompanhar os resultados.

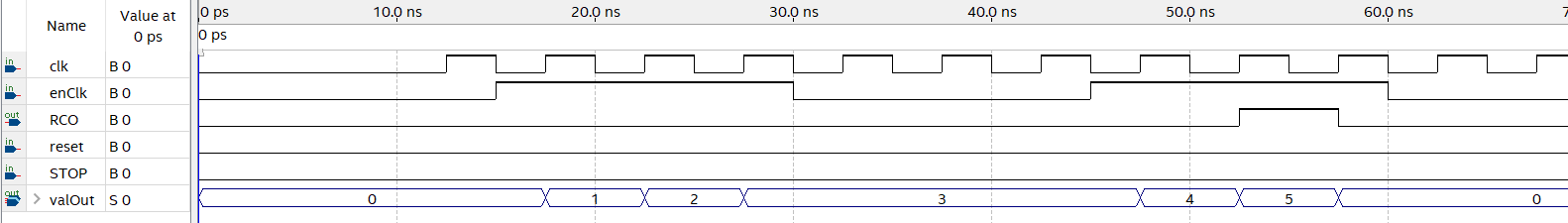


Figura 5: VWF-Counter MAX=5;

**Conclusão**

Em primeiro lugar ao longo da realização do trabalho foram surgindo algumas dificuldades, essas apenas foram possíveis de ser superadas através de uma análise intensiva do código. A terceira fase não foi efetuada com sucesso primeiramente tentei guardar o alarme num registo e depois mais tarde passar para uma RAM, mas sem sucesso.

Autoavaliação: Apesar do trabalho não estar concluindo devido a problemas na terceira fase, a primeira fase e a segunda fase surgem operacionais, logo a autoavaliação recai sobe 12,5 valores.

Concluindo, este trabalho permitiu-me adquirir mais conhecimentos sobre a programação em VHDL e aprender um pouco sobre o funcionamento dos despertadores e da sua modelação.

**Manual do utilizador**

|  |  |
| --- | --- |
| KEY/HEX |  |
| HEX0 | Minutos, unidade |
| HEX1 | Minutos, dezenas |
| HEX2 | Horas, unidades |
| HEX3 | Horas, dezenas |
| HEX4 | Dia da semana |
| HEX7 | Modo de Funcionamento |
| KEY0 | Incrementar 1 unidade |
| KEY1 | Decrementar 1 unidade |
| KEY2 | Seleção de dígitos |
| KEY3 | Seleção de alarmes |

|  |  |
| --- | --- |
| SW(0) E SW(1) | Modo de Configuração |
| “00” | Normal de contagem |
| “01” | Acerto de horas/dias da semana |
| “11” | Programação da função despertador |

LEDG(8) – INDICAÇÃO DE SEGUNDOS

